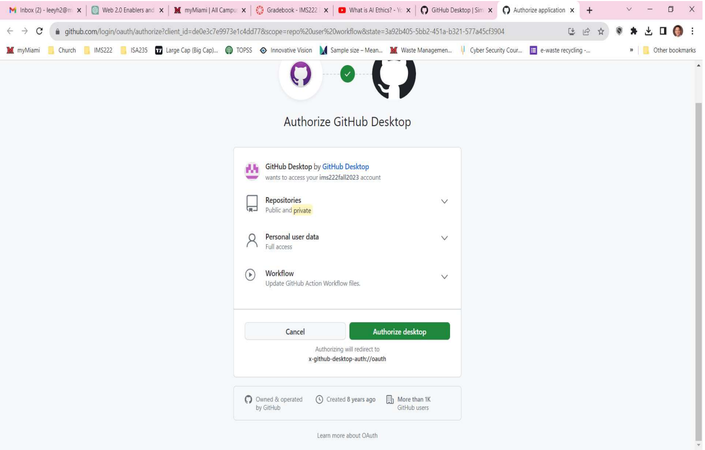
1. Download GitHub Desktop at: https://desktop.github.com/
2. Install it. When installed, the app will ask you to sign into your GitHub account. If you are new to GitHub, create your account.
3. Click Authorize GitHub Desktop button



github.com/login/oauth/authorize?client_id=de0e3c7e9973e1c4dd77&scope=repo%20user%20workflow&state=3a92b405-5bb2-451a-b321-577a45cf3904

Authorize GitHub Desktop

GitHub Desktop by GitHub Desktop
wants to access your ims222fall2023 account

Repositories
Public and private

Personal user data
Full access

Workflow
Update GitHub Action Workflow files.

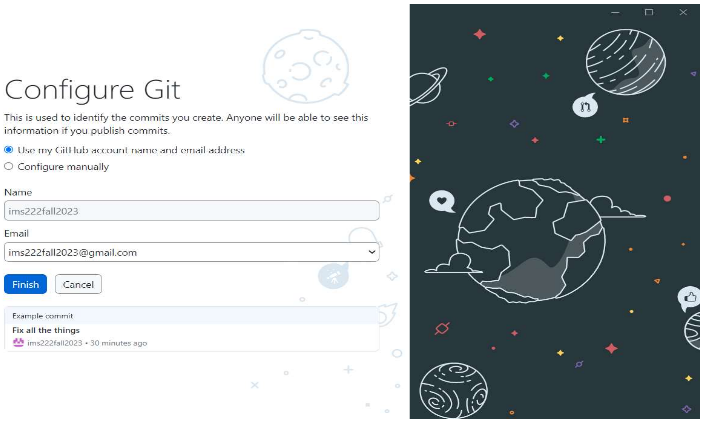
Cancel | Authorize desktop

Authorizing will redirect to
x-github-desktop-auth://oauth

Owned & operated by GitHub | Created 8 years ago | More than 1K GitHub users

Learn more about OAuth

4. then Config Git and click Finish



Configure Git

This is used to identify the commits you create. Anyone will be able to see this information if you publish commits.

● Use my GitHub account name and email address
○ Configure manually

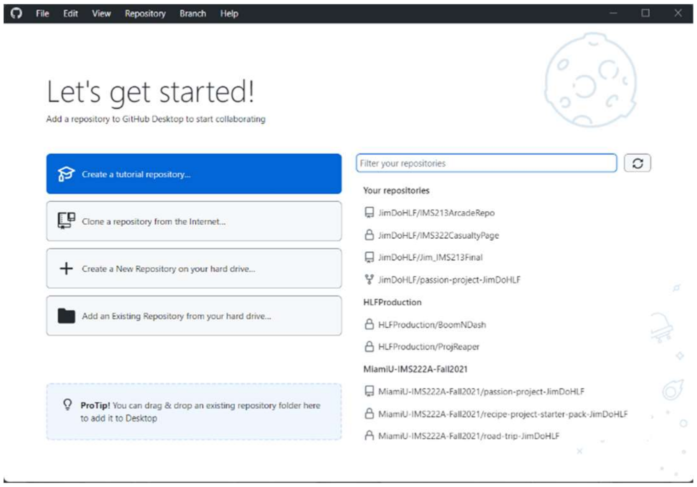
Name
ims222fall2023

Email
ims222fall2023@gmail.com

Finish | Cancel

Example commit
Fix all the things
ims222fall2023 • 30 minutes ago

5. First look at the Desktop app (without active repository)



Let's get started!
Add a repository to GitHub Desktop to start collaborating

Create a tutorial repository...
Clone a repository from the Internet...
Create a New Repository on your hard drive...
Add an Existing Repository from your hard drive...

ProTip! You can drag & drop an existing repository folder here to add it to Desktop

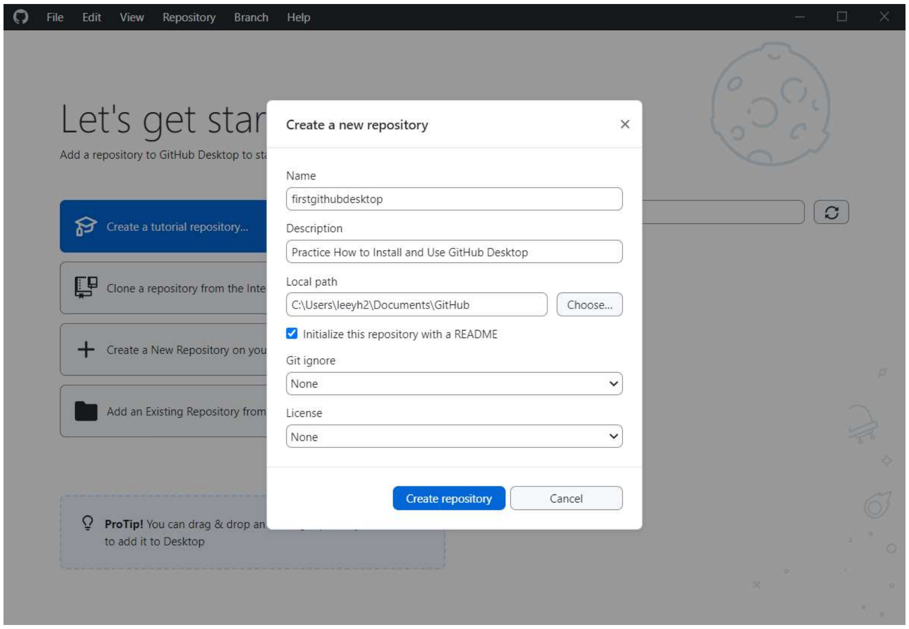
Filter your repositories

Your repositories
JimDoHLF/IMS213ArcadeRepo
JimDoHLF/IMS322CasualtyPage
JimDoHLF/Jim_IMS213Final
JimDoHLF/passion-project-JimDoHLF

HLFProduction
HLFProduction/BoomNDash
HLFProduction/ProjReaper

MiamiU-IMS222A-Fall2021
MiamiU-IMS222A-Fall2021/passion-project-JimDoHLF
MiamiU-IMS222A-Fall2021/recipe-project-starter-pack-JimDoHLF
MiamiU-IMS222A-Fall2021/road-trip-JimDoHLF

6. Click Create a New Repository on Your Hard Drive option and fill out the form. You can either find an existing repository or clone a repository from the Internet (including your own repository)


File
Edit
View
Repository
Branch
Help
Let's get star
Add a repository to GitHub Desktop to st
Create a tutorial repository...
Clone a repository from the Inte
Create a New Repository on you
Add an Existing Repository from
Choose...
Create repository
Cancel

Create a new repository

Name: firstgithubdesktop

Description: Practice How to Install and Use GitHub Desktop

Local path: C:\Users\leeyh2\Documents\GitHub

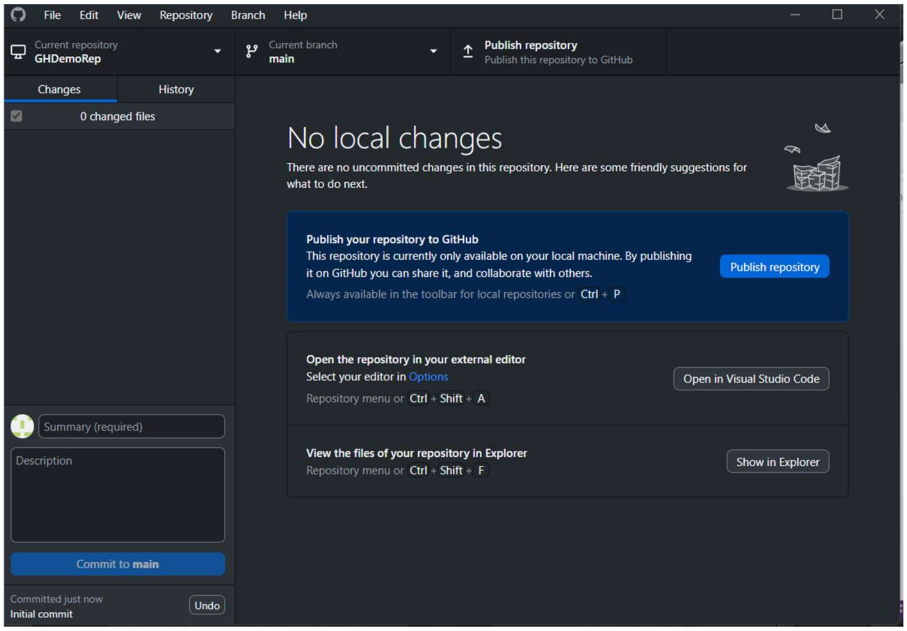
☑ Initialize this repository with a README

Git ignore: None

License: None

ProTip! You can drag & drop an
to add it to Desktop

7. A look at the GitHub Desktop interface:



0 changed files

No local changes

There are no uncommitted changes in this repository. Here are some friendly suggestions for what to do next.

Publish your repository to GitHub
This repository is currently only available on your local machine. By publishing it on GitHub you can share it, and collaborate with others.
Always available in the toolbar for local repositories or Ctrl + P

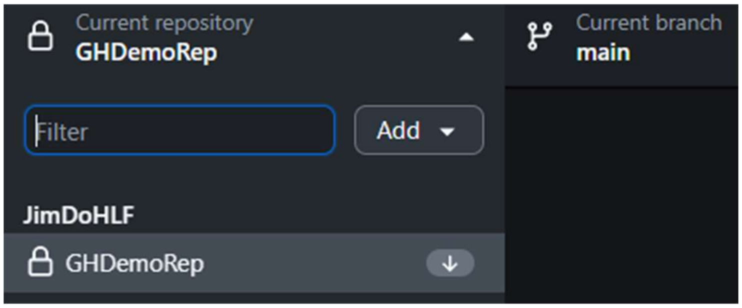
Open the repository in your external editor
Select your editor in Options
Repository menu or Ctrl + Shift + A

View the files of your repository in Explorer
Repository menu or Ctrl + Shift + F

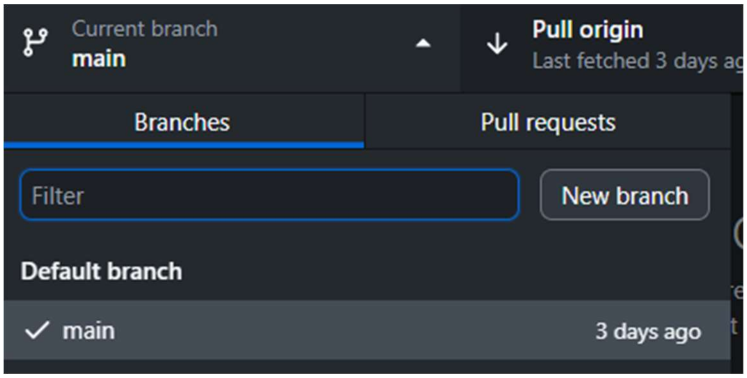
Committed just now
Initial commit

8. When working with your local repository: work in your IDE of choice. Files are available offline on your computer, meaning you can work anytime from your computer.
9. Navigate the desktop interface.

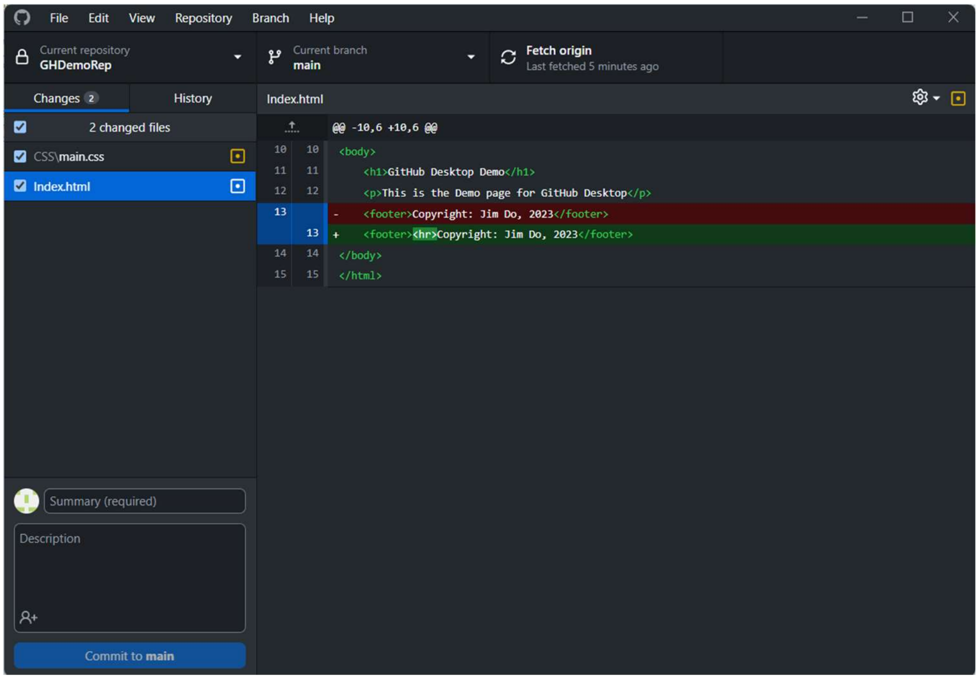
Repository



Branch



10. Change history

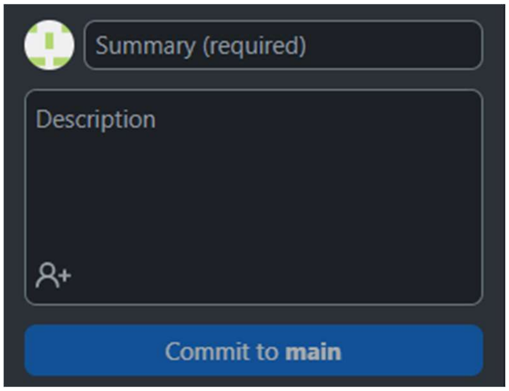


Index.html

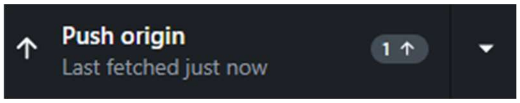
```
@@ -10,6 +10,6 @@
  <body>
      <h1>GitHub Desktop Demo</h1>
      <p>This is the Demo page for GitHub Desktop</p>
-     <footer>Copyright: Jim Do, 2023</footer>
+     <footer><hr>Copyright: Jim Do, 2023</footer>
  </body>
  </html>
```

Commit, push/pull, fetch origin.

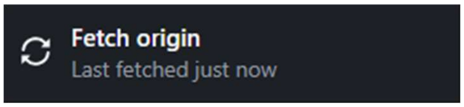
Commit box: summary title and description, as well as commit branch



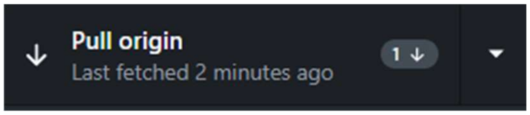
Push: Push the new commit to the repository



Fetch: Check for new updates online

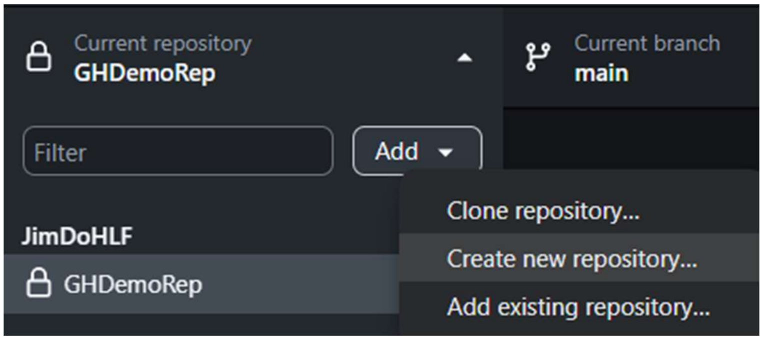


Pull: Get new commit from repository online



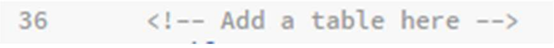
Hands-on Activity:

1. Download and setup GitHub Desktop (following the steps above)
2. Log in to your account and create a new local repository.



3. Download GHDemo.zip from this week's Module.
4. Put and unzip the demo inside the local repository.
5. Play around with the Repository:
   a. Initial Commit: Commit the repository and push to your GitHub
   - Name the commit: "init commit", write some descriptions in the box

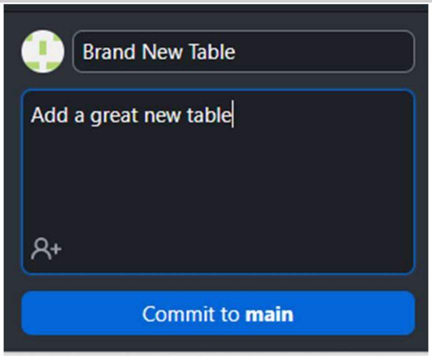
   b. Working on the Repository:
   - Open up *index.html* file using Bracket
   - Add a new table into <body>, under the comment:



```
<!-- Add a table here -->
```

```
<table>
   <tr>
      <td>Hello</td>
      <td>World</td>
   </tr>
   <tr>
      <td>Welcome</td>
      <td>Friend</td>
   </tr>
</table>
```

- Check your GitHub Desktop, make a commit, and push to online repository.



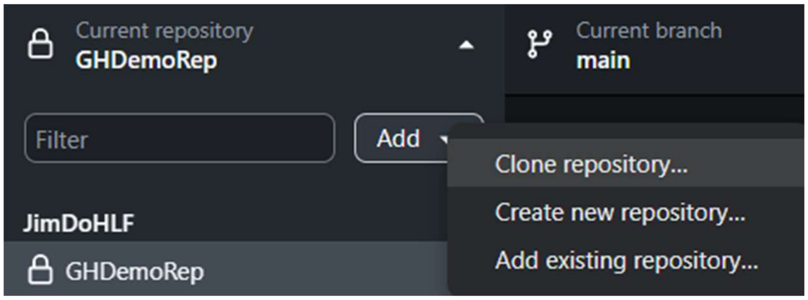
Brand New Table

Add a great new table

c. Pulling online repository from computer

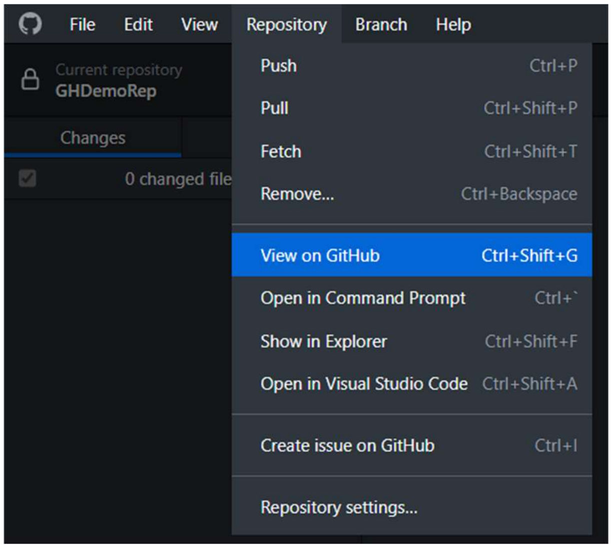
- Getting a new repository:

Since we created our repository offline, it is stored in the list already. However, to create the offline repository from an existing online repository:

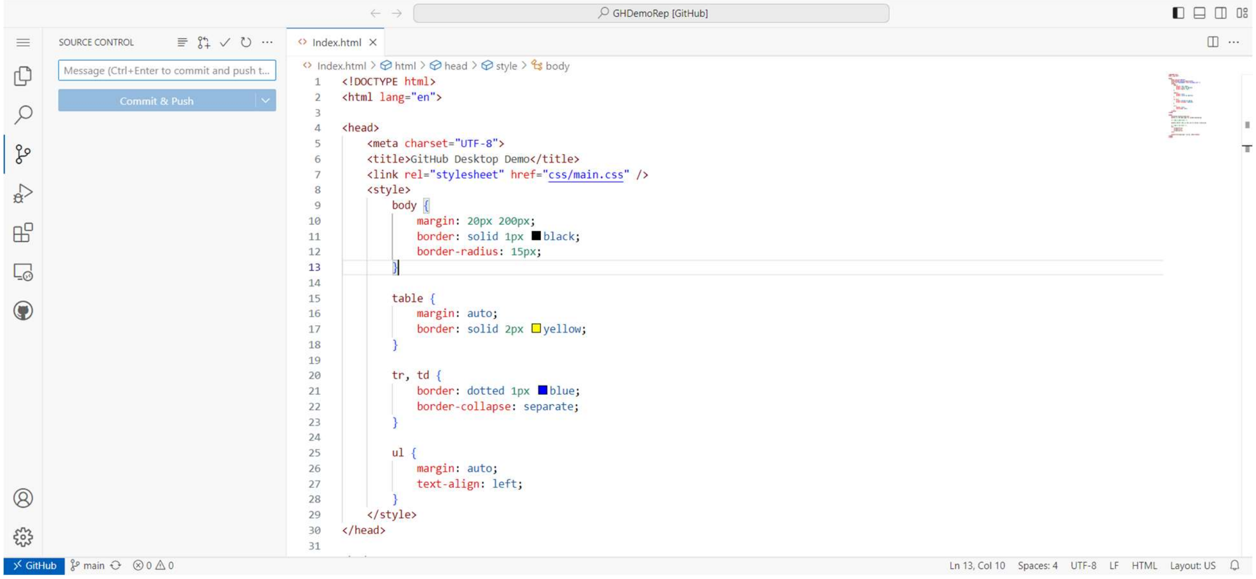


- Go to your GitHub website, find your repository

+ A quick way to do this is by navigating the *Repository* tab. There are several functions (and keyboard shortcuts) you can use. Here, we use *View on GitHub* to directly access the repository online (might require signing in to GitHub website again)



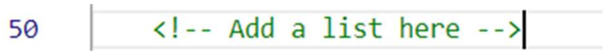
- On GitHub website, press the period button [.] to open web editor.



```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <title>GitHub Desktop Demo</title>
    <link rel="stylesheet" href="css/main.css" />
    <style>
        body {
            margin: 20px 200px;
            border: solid 1px black;
            border-radius: 15px;
        }

        table {
            margin: auto;
            border: solid 2px yellow;
        }

        tr, td {
            border: dotted 1px blue;
            border-collapse: separate;
        }

        ul {
            margin: auto;
            text-align: left;
        }
    </style>
</head>

```

The editor is a version of Visual Studio, but you don’t need to install Visual Studio (any version) on your computer to function.

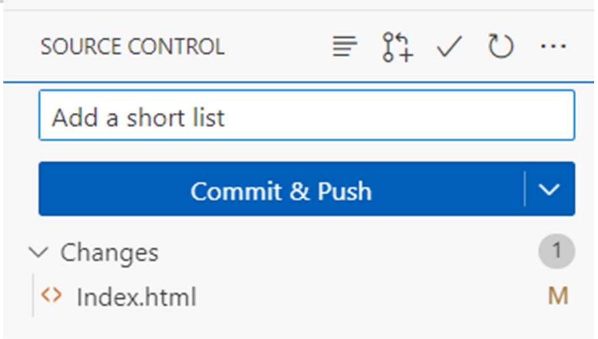
- Edit the *index.html* file, add a new list under the comment:



```
<!-- Add a list here -->
```

```
<ul>
   <li>Earth</li>
   <li>Mars</li>
   <li>World</li>
</ul>
```

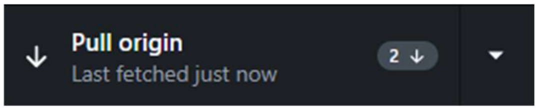
- On the left-hand side, type in a message (commit title) and press *Commit & Push.*



Add a short list

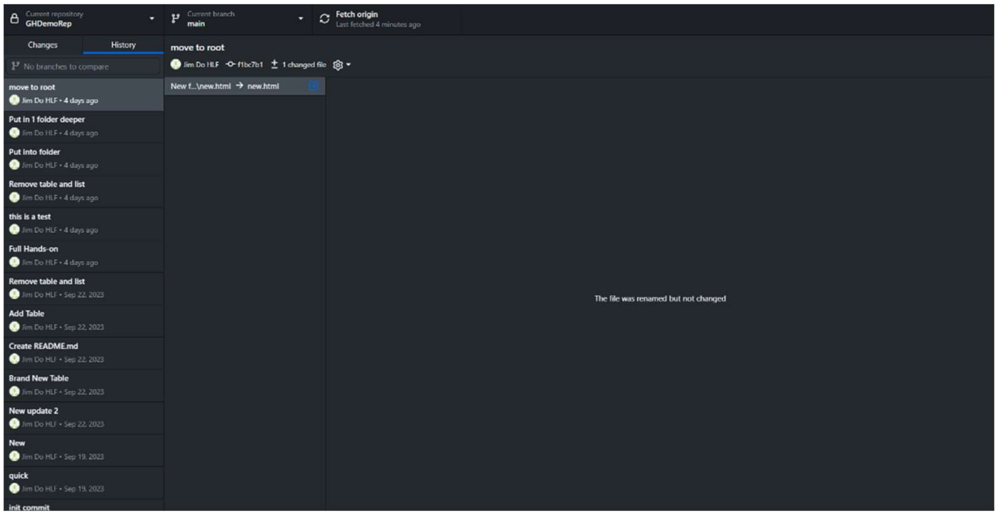
- You have committed some changes outside of your local offline repository (controlled by GitHub Desktop). Open GitHub Desktop:

+ Changes might not be detected automatically. Press *Fetch Origin* and wait until the button changes to *Pull Origin* (detect new commit on the same branch in the same repository):



- Now the new changes made online have been updated into your local repository. This doesn't only include changes you made online, but also any commits made by your teammates.

BONUS: Check branch history:



move to root
Jim Do HLF • 4 days ago

Put in 1 folder deeper
Jim Do HLF • 4 days ago

Put into folder
Jim Do HLF • 4 days ago

Remove table and list
Jim Do HLF • 4 days ago

this is a test
Jim Do HLF • 4 days ago

Full Hands-on
Jim Do HLF • 4 days ago

Remove table and list
Jim Do HLF • Sep 22, 2023

Add Table
Jim Do HLF • Sep 22, 2023

Create README.md
Jim Do HLF • Sep 22, 2023

Brand New Table
Jim Do HLF • Sep 22, 2023

New update 2
Jim Do HLF • Sep 22, 2023

New
Jim Do HLF • Sep 19, 2023

quick
Jim Do HLF • Sep 19, 2023

init commit

move to root
Jim Do HLF f1bc7b1 1 changed file

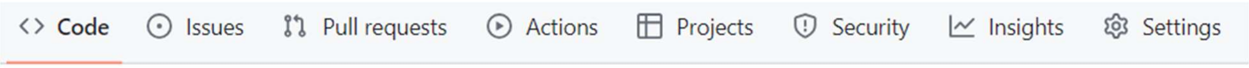
New f...\new.html → new.html

The file was renamed but not changed

NOTE: GitHub Desktop is an excellent tool for working in the local environment (offline, using your local IDE), but it lacks many functionalities available on github.com

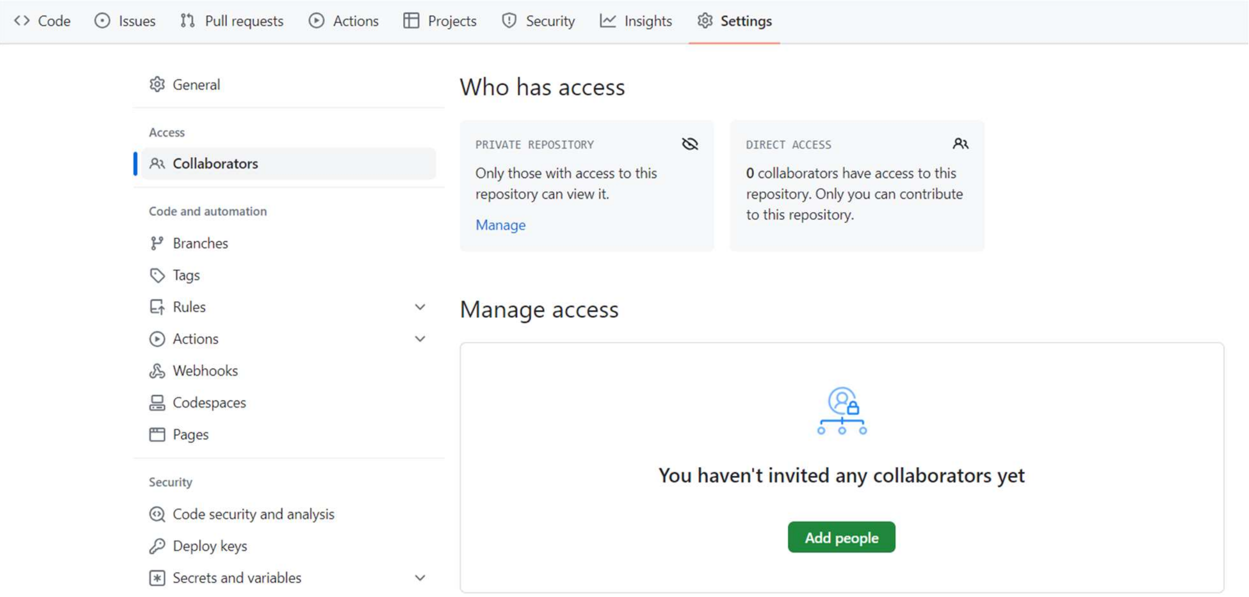
For other functionalities including website hosting, reviewing commits and pull requests, use the website. Otherwise, to work on the files, you can utilize GitHub Desktop.

GitHub Online Environment



- So far, we have been working on GitHub Desktop (local environment).
- GitHub Desktop primarily works within the Code tab, as well as features from some other tabs related to file management.
- However, major features (including settings, pull requests, etc. should, and can only, be done on GitHub online environment (at github.com).
- Let's quickly look through the major features you might engage with:

1. Collaborators



Who has access

PRIVATE REPOSITORY

Only those with access to this repository can view it.

DIRECT ACCESS

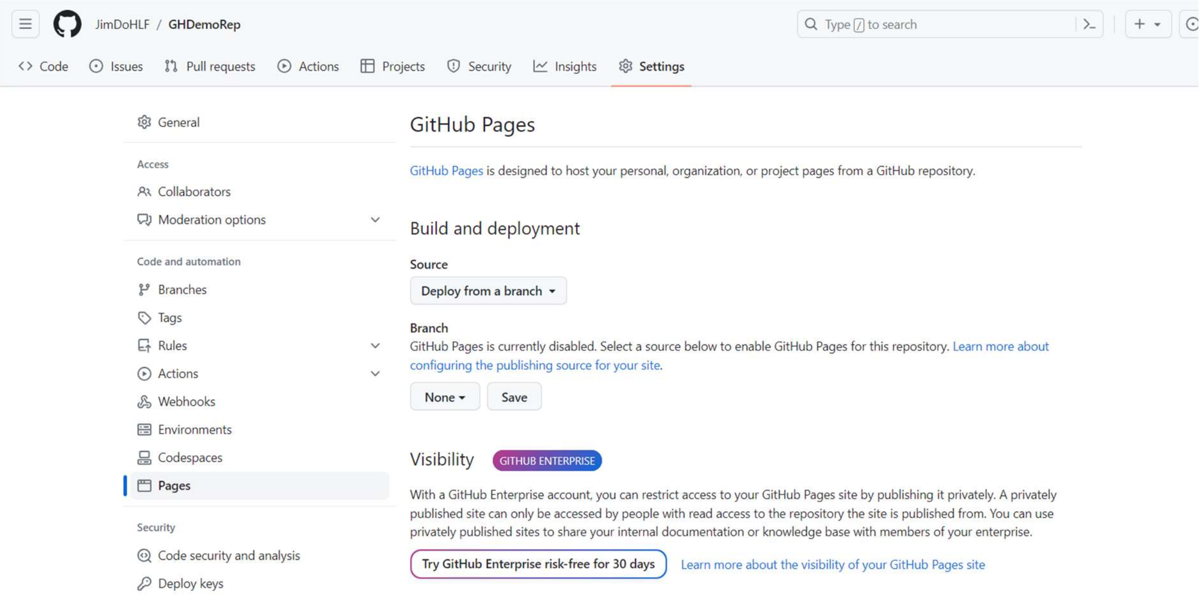
0 collaborators have access to this repository. Only you can contribute to this repository.

Manage access

You haven't invited any collaborators yet

- Here you can manage who gets access to the repository (visibility):
  - Private: Only available for selected collaborators
  - Public: Everyone can access the repository
- Add Collaborators:
  - Allow special permissions to the repository beyond the visibility used
  - Have access to edit the files in branches, but not full access like the owner (including changing settings, etc.)

2. Page



GitHub Pages

GitHub Pages is designed to host your personal, organization, or project pages from a GitHub repository.

Build and deployment

Source

Deploy from a branch

Branch

GitHub Pages is currently disabled. Select a source below to enable GitHub Pages for this repository. Learn more about configuring the publishing source for your site.

None ▾ Save

Visibility GITHUB ENTERPRISE

With a GitHub Enterprise account, you can restrict access to your GitHub Pages site by publishing it privately. A privately published site can only be accessed by people with read access to the repository the site is published from. You can use privately published sites to share your internal documentation or knowledge base with members of your enterprise.

Try GitHub Enterprise risk-free for 30 days Learn more about the visibility of your GitHub Pages site

- Build and deploy your files to create a website.
- Choose a branch, save, and wait for GitHub to setup the website.